Robust Camera Pose Refinement and Rapid SfM for Multiview Aerial Imagery—Without RANSAC

Hadi Aliakbarpour, Kannappan Palaniappan, and Guna Seetharaman

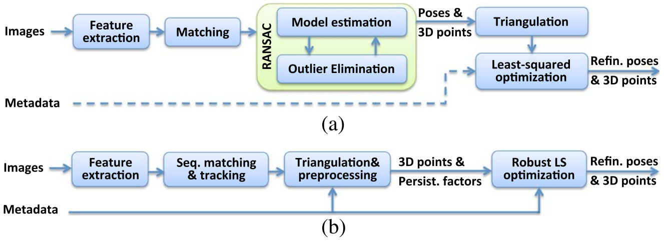
***Abstract*—A robust camera pose refinement approach for sequential wide-area airborne imagery is proposed in this letter. Image frames are sequentially acquired, and with each frame, its corresponding position and orientation are approximately available from airborne platform inertial measurement unit and GPS sensors. In the proposed structure from motion (SfM) approach the available approximation of camera parameters (from low-certainty sensors) is directly used in an optimization stage. The putative matches obtained from the sequential matching paradigm are also directly used in the optimization with no early stage filtering (e.g., no RANSAC). A robust function is proposed and used to deal with outliers (mismatches). The full pipeline has been run over a set of wide-area motion imagery data collected by an airplane flying over different cities in the U.S. The results prove the power and efficiency of the proposed pipeline. Effectiveness of the proposed robust function is compared with some popular robust functions such as Cauchy and Huber using synthetic data.**

***Index Terms*—Bundle adjustment (BA), exterior orientation (EO), structure from motion (SfM), 3-D reconstruction and multiview stereo (MVS).**

I. Introduction

BUNDLE adjustment (BA) is an essential part of using structure from motion (SfM) and multiview stereo (MVS) computer vision algorithms for 3-D reconstruction using a set of 2-D images. In applications such as 3-D reconstruction, aerial photogrammetry, and computer vision, it is essential to refine the noisy camera pose measurements in order to enable accurate processing of the imagery data. BA is the most popular solution and a gold standard [1], [2] to obtain precise camera poses. It receives initial estimates of camera poses and minimizes the errors based on some cost functions [3]. Despite many reports presented in this old area of research, BA is still a bottleneck in related applications.

Mostly, initial camera poses (inputs to BA) are obtained by applying a RANSAC-based model estimation algorithm (e.g., the five-point algorithm [4]–[6]). However, nowadays in aerial imagery systems, these parameters are often available and known as *a priori* which can be directly measured with onboard sensors [GPS and inertial measurement unit (IMU)]. Nevertheless, these parameters are too noisy [7] and must be refined before being used in the downstream processing stages (e.g., 3-D reconstruction [6], [8], [9]).



(a)

(b)

Fig. 1. Conventional and proposed SfM (BA pipelines). (a) Conventional SfM. Camera poses and outliers are simultaneously estimated using RANSAC. Metadata sometimes is used as an extra constraint in optimization. (b) BA4W; proposed SfM (BA pipeline). Metadata is directly used. No model estimation, no explicit outlier elimination, and no RANSAC.

Throughout this letter, wherever the term “BA pipeline” is used, it refers to an end-to-end BA system (or SfM) whose inputs are raw images and outputs are refined camera poses and 3-D point cloud. Likewise, when the term “BA” is used, it will refer to just the optimization stage where *initial* camera poses and point cloud are already available.

This letter proposes a new SfM (BA pipeline) to refine noisy camera parameters available from platform sensors. In the proposed approach, to be called *BA4W*, approximate sensor measurements are directly used as initial values for BA. Moreover, we do not apply any early-stage filtering method (e.g., no RANSAC nor its variations) to eliminate outliers. A conventional SfM and the proposed one are depicted in Fig. 1. More precisely, the contributions of this letter include the following.

1) We show that approximate camera parameters available from low-precision sensors in an airborne platform (GPS and IMU sensors) can be directly deployed in a BA as initial values (and not as extra constraints [10]–[12]), provided that a proper robust function is used. It will be shown that there is no need to apply any camera pose estimation method (e.g., the five-point algorithm [4], [7], [12]). Neither it will not be needed to apply a filtering method such as extended Kalman filter (EKF) [10], [11] before feeding noisy sensor measurements into the optimization engine.
2) We demonstrate that the putative matches obtained from a sequential matching paradigm can be directly used as observations and initial 3-D points in the BA. We show that there is no need to eliminate outliers before feeding the putative matches to the optimization engine.

Manuscript received March 20, 2015; revised May 24, 2015 and June 14, 2015; accepted July 9, 2015. Date of publication August 11, 2015; date of current version October 27, 2015.This work was supported in part by the U.S. Air Force Research Laboratory.

H. Aliakbarpour and K. Palaniappan are with the Department of Computer Science, University of Missouri, Columbia, MO 65211 USA (e-mail: hd.akbarpour@gmail.com).

G. Seetharaman is with the Air Force Research Laboratory, Rome, NY 13441 USA.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/LGRS.2015.2457299

For instance, there will be no need to use RANSAC or any other explicit outlier filtering mechanism.

3) An adaptive robust function is proposed to deal with outliers within the optimization stage. It uses information about the quality of the 2-D feature correspondences and is shown to be superior when compared to other popular robust error norms such as Cauchy and Huber.

Background: In the computer vision community, camera parameters are known as *intrinsic* and *extrinsic*. In photogrammetry, the same parameters are known as *interior* and *exterior* parameters. Having precise values of these parameters is very crucial for relevant applications (e.g., 3-D reconstruction). BA is considered as the gold standard for refinement [1], [2], [13] of camera parameters and is a widely studied problem in computer vision, robotics, and photogrammetry dating back more than three decades [3], [13]. A comprehensive introduction to BA can be found in [3], which covers a wide spectrum of topics involved in BA. Due to recent interest in large-scale 3-D reconstruction from consumer photographs as well as aerial imagery, there have been renewed interests in making BA robust, stable, and accurate [5], [14]–[16]. Recently, several BA methods have been proposed, such as sparse BA [17], [18], incremental BA [10], and parallel BA [19], [20].

There have been many reports presenting the use of GPS and IMU measurements for refining camera parameters. However, to our best knowledge, so far such measurements have been mostly used as complementary values and just together with other pose estimation methods through essential matrix estimation (in computer vision) [10], [11] or resectioning in photogrammetry. For example, in [8], [10], [11], and [21], available GPS and IMU measurement are fused with SfM approach using an EKF or/and as extra constraints in BA. An SfM method, called *Mavmap*, is proposed in [12], which leverages the temporal consistency of aerial images and availability of metadata to speed up the performance and robustness. In [12], VisualSFM [19] has been considered as the most advanced and widely used system for automated and efficient 3-D reconstruction from 2-D images. However, as stated in [12], it is not efficient for aerial imagery and also has no integration of IMU priors.

II. Building Feature Tracks

In persistent aerial imagery (WAMI), flights have continuous circular trajectories, yielding temporal consistency to the image sequence. By leveraging the temporal consistency of the images and using them as a prior information, we reduce the time complexity of matching from $O(N_c^2)$ to $O(N_c)$ while not compromising the quality of pose refinement results (N_c is the number of cameras). This is similar to what has been recently proposed in [12]. In this method, interest points are extracted from each image using a proper feature extraction method. Starting from the first frame, for each two successive image frames, the descriptors of their interest points are compared. While successively matching them along the sequence, a set of feature *tracks* is generated. A track indicates that a potentially unique 3-D point in the scene has been observed in a set of image frames. The minimum length for a track is two, and it ideally can go up to N_c. N_o will be used to represent the total number of 2-D feature points (image observations), and $N_{3\mathrm{D}}$ represents the total number of tracks (3-D points). Normally, the tracks are just used as a way to associate a set of feature points together from which a 3-D point is estimated in the scene. In addition to this, we consider the length of a track j as a factor of persistency γ_j and use it in the optimization. Indeed, γ_j is equivalent to the number of persistent covisibility of jth 3-D point in the sequence. The intuition is that a detected feature point is more reliable if detected over a longer period of time in a sequence. It is analogous to say that a 3-D point estimated from a shorter track is more vulnerable to be a spurious point, which can adversely affect the optimization. Therefore, with each track of features, a persistency factor is assigned and stored. After building all tracks, the expected value of all of the persistency factors $\mu = (1/N_{3\mathrm{D}}) \sum_{j=1}^{N_{3\mathrm{D}}} \gamma_j$ and their standard deviation (std) σ are calculated. These two statistical factors will be used together with the persistency factors γ_j $(j = 1, \ldots, N_{3\mathrm{D}})$ within the optimization stage (see Section III-B).

III. Robust Pose Refinement

A. BA Formulation

BA refers to the problem of jointly refining camera parameters and 3-D structure in an optimal manner. Given a set of N_c cameras, with possibly different poses (translations and orientations) and $N_{3\mathrm{D}}$ points, the BA is done by minimizing the sum of squared reprojection errors

$$\min_{\mathbf{R}_i,\mathbf{t}_i,\mathbf{X}_j} \sum_{i=1}^{N_c} \sum_{j=1}^{N_{3\mathrm{D}}} \left\| \mathbf{x}_{ji} - g(\mathbf{X}_j, \mathbf{R}_i, \mathbf{t}_i, \mathbf{K}_i) \right\|^2 \tag{1}$$

where $\mathbf{R}_i$, $\mathbf{t}_i$, and $\mathbf{K}_i$ are, respectively, the rotation matrix, translation vector, and calibration matrix of the ith camera, $\mathbf{X}_j$ is a 3-D point from the structure, and $\mathbf{x}_{ji}$ is the image coordinates (observation) of $\mathbf{X}_j$ in camera i. Here, $g(\mathbf{X}_j, \mathbf{R}_i, \mathbf{t}_i, \mathbf{K}_i)$ is a projection model which maps a 3-D point $\mathbf{X}_j$ onto the image plane of camera i using its related extrinsic ($\mathbf{R}_i$ and $\mathbf{t}_i$) and intrinsic parameters ($\mathbf{K}_i$) and is defined as

$$g(\mathbf{X}_j, \mathbf{R}_i, \mathbf{t}_i, \mathbf{K}_i) \sim \mathbf{K}_i \left[\mathbf{R}_i | \mathbf{t}_i\right] \mathbf{X}_j. \tag{2}$$

Usually, $g(\mathbf{X}_j, \mathbf{R}_i, \mathbf{t}_i, \mathbf{K}_i) = \mathbf{x}_{ji}$ is not satisfied due to noisy parameters, and a statistically optimal estimate for the camera parameters and 3-D points is desired. This L_2 minimization of the reprojection error involves adjusting the bundle of rays between each camera center and the set of 3-D points which is a nonlinear constrained optimization problem. It is equivalent to finding a maximum likelihood solution assuming that the measurement noise is Gaussian, and we refer to [3] and [13] for more details. There exist various methods to solve the aforementioned nonlinear least squares problem. Implicit trust region methods and, in particular, Levenberg–Marquardt (LM) methods are well known in the BA literature [15], [17].

B. Adaptive Robustifier

Automatic selection of 2-D point correspondences (tie points) is arguably known as one of the most critical steps in image-based multiview reconstruction [22]. Feature correspondences are usually contaminated by outliers, which is wrong data association. Mostly, pose refinement or SfM techniques in literature use initial estimates and then perform a refinement

TABLE I

DATA SET SPECIFICATIONS AND TIMINGS FOR INDIVIDUAL PROCESSING STEPS (PER IMAGE) AND OVERALL WITH COMPARISON TO TWO OTHER APPROACHES. N_c, N_o, AND N_{3D} STAND FOR THE "NUMBER OF CAMERAS," "NUMBER OF OBSERVATIONS," AND "NUMBER OF 3-D POINTS," RESPECTIVELY. THE TOTAL TAKEN TIME AND PER IMAGE SPEEDS ARE PRESENTED FOR BA4W (OUR METHOD), FOR VISUALSFM [19], AND FOR MAVMAP [12]. THE SPEED-UP FACTOR FOR OUR METHOD VERSUS EACH OF THE TWO OTHER METHODS IS PRESENTED AND HIGHLIGHTED. IN AVERAGE, BA4W IS 77 TIMES FASTER THAN VISUALSFM AND 22 TIMES FASTER THAN MAVMAP

Dataset specification			BA4W		Time: BA4W						Time: VisualSfM		Speed-up	Time: Mavmap		Speed-up
					Per stage				Total				BA4W=			BA4W=
Dataset	Image size	N_c	N_o	N_{3D}	Iter.	Feat. + Track.	Triang.	Optim.	Whole seq.	Per image	Whole seq.	Per image	×VSFM	Whole seq.	Per image	×Mavmap
PIXHAWK [12]	752×480	300	254,046	45,845	25	11s	5s	19s	35s	0.12s	*NA*	*NA*	*NA*	19m	3.80s	32.57
Four hills	6048×4032	100	262,828	80,661	36	42s	8s	16s	1m + 06s	0.66s	36m	21.60s	32.73	34m	20.04s	30.91
Columbia	6600×4400	202	655,593	115,897	10	235s	13s	15s	4m + 23s	1.30s	*NA*	*NA*	*NA*	60m	17.82s	13.69
Albuquerque	6600×4400	215	668,000	141,559	30	223s	15s	35s	4m + 33s	1.27s	4h + 25m	73.95s	58.24	1h + 07m	18.70s	14.72
Berkeley	6600×4400	220	683,123	138,743	24	185s	16s	43s	4m + 04s	1.11s	4h + 40m	76.37s	68.85	1h + 03m	17.80s	15.24
LA	6600×4400	351	1,115,603	207,391	10	230s	23s	39s	4m + 52s	0.83s	8h + 05m	78.29s	99.66	1h + 43m	17.78s	21.37
AlbuquerqueFull	6600×4400	1,071	3,473,122	603,119	30	467s	63s	222s	12m + 32s	0.70s	26h + 36m	85.37s	127.34	5h + 21m	17.98s	25.61

which generally happens by iteratively applying the LM algorithm [3] on the initial camera parameters and 3-D points. LM is highly sensitive to the presence of outliers in the input data [22]. Mismatches can cause problem for the standard least squares approach, and as stressed in [23], even a single mismatch can adversely affect the entire result. They can easily lead to suboptimal parameter estimation or inability of the optimization algorithm to obtain a feasible solution [22], [24]. This is even more problematic for the high-resolution WAMI, where a number of potential correspondences are enormous. Generally, outliers are explicitly excluded from the putative matches in very early stages and much before applying an optimization. For example, in computer vision, a relative camera motion estimating algorithm is applied (e.g., the five-point algorithm [4]) in which simultaneously initial camera parameters are estimated while explicitly detecting and eliminating outliers (which happens mostly through different variations of RANSAC). Here, we show that there is no need to apply any explicit outlier elimination algorithm. We show that the choice of a proper robust function is very crucial to achieve robustness in the optimization when the initial parameters are too noisy and outliers were not explicitly eliminated.

In Section II, it was motivated that a feature persistently observed in successive frames along the airplane trajectory is less likely to produce a spurious 3-D point. Inspired from the Cauchy robust function [3], we propose a robust function which integrates this theory by using the statistics calculated in the feature-tracking stage

$$\rho_j(s_j, \gamma_j, \mu, \sigma) = \left(\frac{\gamma_j}{\mu+\sigma}\right)^2 \log\left(1 + \left(\frac{\mu+\sigma}{\gamma_j}\right)^2 s_j^2\right) \quad (3)$$

where s_j denotes the residual of the jth 3-D point, γ_j stands for its persistency factor, and μ and σ are, respectively, the mean and std of the persistency factors of the whole track population. For each individual residual, its persistency factor is normalized by being divided to the sum of the mean and std of the population, and the result is used as a weight in the robust function. A larger persistency factor for a track is seen analogous to a longer covisibility period over the image sequence (higher persistency on their continued observation over the trajectory). Residuals belonging to a track with a longer covisibility period (larger γ_j) are favored over residuals with shorter covisibility period (smaller γ_j). Thus using (3), (1) can be expressed as

$$\min_{\mathbf{R}_i, \mathbf{t}_i, \mathbf{X}_j} \sum_{i=1}^{N_c} \sum_{j=1}^{N_{3D}} \rho\left(\left\|\mathbf{x}_{ji} - g(\mathbf{X}_j, \mathbf{R}_i, \mathbf{t}_i, \mathbf{K}_i)\right\|, \gamma_j, \mu, \sigma\right). \quad (4)$$

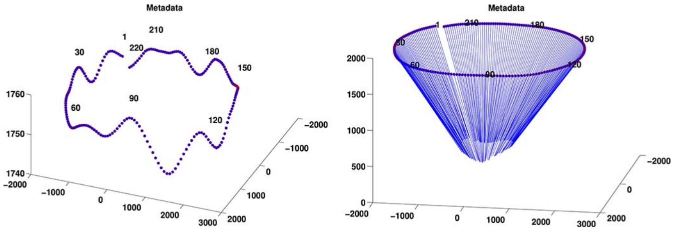
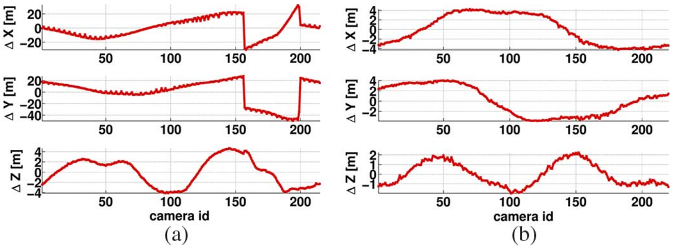


Fig. 2. Camera trajectory for the Berkeley data set. (Left) Positions. (Right) Looking directions.

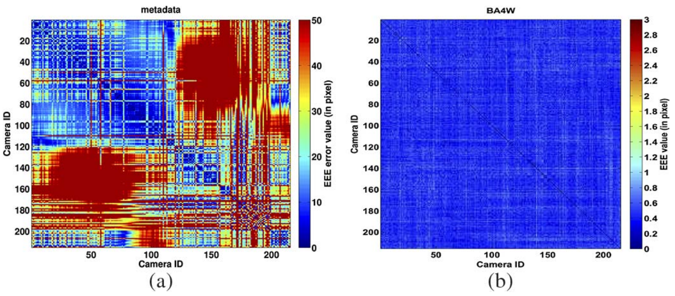


(a) (b)

Fig. 3. Difference between camera positions of metadata and BA4W output. They basically indicate how much the camera positions have been corrected after BA. (a) Albuquerque. (b) Berkeley.

IV. EXPERIMENTS

The BA4W pipeline has been implemented in C++. The used PC has the following specifications: CPU Intel Xeon 5650, 2.66 GHz, 12 cores (24 threads), 24-GB RAM, and nVidia GTX480/1.5 GB as the GPU. SIFT-GPU [25] has been used for fast feature extraction. We used Ceres Solver library [26] as a framework for nonlinear least squares estimation. Schur's complement, Cholesky factorization, and LM algorithm were used for trust region step computation. The WAMI data sets (see Table I) were collected from platform sensors of an airplane flying over different areas in the U.S. including Albuquerque downtown, Four hills (Albuquerque), Los Angeles, Berkeley, and Columbia. In addition to the images, camera poses were measured by IMU and GPS sensors (referred to as *metadata*). The BA4W pipeline has been run on each data set. A nonlinear triangulation algorithm [13] was used to estimate 3-D points. Also, the persistency factors of the tracks and their related statistics were computed, and used in the optimization. The camera positions and their looking directions corresponding to the Berkeley data set are plotted in Fig. 2. Fig. 3 depicts the amount of corrections accomplished by BA4W on the camera positions. As one can see, in Albuquerque, the correction value for one of the axes $(\mathbf{Y})$ is about 40 m, which indicates how noisy the metadata were for this data set.



(a) (b)

Fig. 4. Representation of error values between each pair of cameras (a cell in the plot) within one of the data sets (Albuquerque), after and before BA. The plotted ϵ_{ij} values were truncated once bigger than the spectrum maximum value. Notice that, for the plots corresponding to the original metadata and refined ones, different scales are used for better presentation. (a) $(\mu_\epsilon, \sigma_\epsilon) = (39.57, 33.74)$. (b) $(\mu_\epsilon, \sigma_\epsilon) = (0.47, 0.12)$.

Fig. 4 represents the error values between each pair of cameras (a cell in the plot) within the Albuquerque data set before and after BA4W. For each data set, a groundtruth has been generated by manually tracking a set of N_g points over the image sequences. For each pair of cameras (i and j), their mutual pose error ϵ_{ij} is measured by

$$\epsilon_{ij} = \frac{1}{N_g} \sum_{k=1}^{N_g} d(\mathbf{x}_{kj}, \mathbf{F}_{ij}\, \mathbf{x}_{ki}) \tag{5}$$

where $\mathbf{F}_{ij}$ is the fundamental matrix between the image planes of two cameras and directly computed from the camera pose parameters, and d is the Euclidean distance between a groundtruth point $\mathbf{x}_{kj}$ (in image j) and its corresponding epipolar line $\mathbf{F}_{ij}\ \mathbf{x}_{ki}$ from image i. Fig. 4 presents the ϵ_{ij} values plotted in different color spectra. The left and right plots correspond to evaluation using original camera poses (from metadata) and bundle adjusted ones (by BA4W), respectively. The mean and std of all ϵ_{ij} are computed and displayed under each case. As can be seen, BA4W has been quite successful in the pose refinement process despite taking a significantly short running time (see Table I for more details). It is worth mentioning that, after having the camera parameters optimized a dense and colored point cloud can be obtained by applying a proper dense 3-D reconstruction algorithm such as PMVS [27].

We compared our algorithm versus Mavmap [12] as a recent SfM algorithm for sequential aerial images. Mavmap also takes advantage of temporal consistency and availability of metadata. The camera poses in Mavmap are initialized by estimating the essential matrix [4] and applying a RANSAC technique to deal with large amounts of outliers. In addition to Mavmap, VisualSFM [19] as a state-of-the-art incremental SfM has been run on the data sets. We initially ran VisualSFM in its sequential matching mode, where each frame is matched just with its next frame. VisualSFM failed to generate reasonable results by producing several fragments of cameras, and only a fraction of cameras could be recovered, and for the other cameras, it failed. This observation about VisualSFM's performance on sequential aerial images is consistent with what was reported by [12]. Therefore, we ran VisualSFM in its exhaustive matching mode to get reasonable results. Table I shows the comparison results. In these experiments, our algorithm is, in average, 77 times faster than VisualSFM, and 22 times faster than Mavmap. Over the longest data set (AlbuquerqueFull), VisualSFM and Mavmap took about 85 and 18 s, respectively, whereas our method took just 0.7 s to process one frame in average.

Tolerability of BA4W in the Presence of Outliers: Generally, in numerical optimization problems, a robust function may be used in order to mitigate the outlier's effect. Cauchy and Huber [3] are two of such robust functions. In this letter, a new robust function was proposed, which is adaptive by using the persistency factor of the tracks. In this section, the tolerability of the proposed robust function in the presence of highly noisy camera measurements and different amounts of outliers has been studied and compared against the Cauchy and Huber.

For groundtruth, we have generated a set of synthetic data by using the information of a real scenario. First, BA4W was run over the Albuquerque data set (a real data set from Table I). Its outputs, including optimized 3-D points and camera parameters, were taken and considered as the groundtruth. Then, the original image observations from the real data set are discarded. The optimized 3-D points were back-projected onto the 215 image planes using the optimized camera parameters, in order to obtain groundtruth for image observations (invalid observations are discarded). It yields to have a syntactic and accurate groundtruth yet quite similar to a real scenario. Then, a percentage of outliers is added to each track. The image observations for outliers were generated randomly and inserted for randomly chosen cameras in each track. For example, if a track includes 20 cameras and if the outlier percentage was set to 60%, then 30 random camera indices, each with a random image observation, are generated. The 20 original cameras together with the injected 30 random cameras/observations yield to have a total of 50 image observations, where 60% of them are outliers. The contaminated observations (correct image points plus the outliers) were used together with the original metadata (initial noisy camera parameters from the original real data set, not the optimized ones) as inputs to the numerical optimization. Within the optimization, four different situations have been considered: 1) using no robust function; 2) using the Huber; 3) using the Cauchy; and 4) using the proposed robust function. Two metrics are used to evaluate each one's performance: the RMS of reprojection errors and the differences between the recovered camera poses and their groundtruth. Translation errors are computed based on the following equation:

$$\frac{1}{N_c} \sum_{j=1}^{N_c} \|\widetilde{\mathbf{t}}_j - \mathbf{t}_j\| \tag{6}$$

where $\|\cdot\|$ defines the Euclidean norm and $\widetilde{\mathbf{t}}_j$ and $\mathbf{t}_j$, respectively, stand for the BA output and groundtruth related to the jth camera. The norm of the difference/sum of the quaternions is used as the rotation error [28]

$$\frac{1}{N_c} \sum_{j=1}^{N_c} \min\left\{\|\widetilde{\mathbf{q}}_j - \mathbf{q}_j\|, \|\widetilde{\mathbf{q}}_j + \mathbf{q}_j\|\right\} \tag{7}$$

where $\widetilde{\mathbf{q}}_j$ and $\mathbf{q}_j$ are the quaternion representations for the BA output and groundtruth related to the jth camera. The results are presented in Table II. The errors with acceptable values are highlighted in yellow. As can be seen, the BA with no robust function or with Huber has the highest error values even when zero percent of data is contaminated. It means that just a

TABLE II

PERFORMANCE OF OPTIMIZATION IN THE PRESENCE OF DIFFERENT LEVELS OF OUTLIERS. FOUR DIFFERENT ROBUST FUNCTIONS ARE CONSIDERED: NONE, HUBER, CAUCHY, AND OURS. FOR ALL OF THESE EXPERIMENTS, THE NOISY CAMERA PARAMETERS [INDICATED IN FIG. 3(a)] WERE DIRECTLY USED IN BA. THE ERRORS WITH ACCEPTABLE VALUES ARE HIGHLIGHTED IN YELLOW. N_{3D} FOR THESE EXPERIMENTS IS 135 451

%	N_o	RMS				Error in rotation				Error in translation (meters)			
		Non	Hub	Cauchy	BA4W	Non	Huber	Cauch.	BA4W	None	Huber	Cauch.	BA4W
0	639312	4.3	1.6	0.020464	0.008168	0.0236	0.0502	0.0007	0.0008	251.99	235.68	2.95	0.96
5	672960	3.1	5.7	0.019825	0.010960	0.0528	0.0583	0.0008	0.0008	482.89	283.22	2.88	1.57
10	710346	11.5	4.7	0.032561	0.018892	0.0717	0.0656	0.0004	0.0005	467.98	406.01	5.00	2.82
15	752131	4.6	8.4	0.017211	0.016437	0.0987	0.1401	0.0004	0.0005	562.15	829.79	2.46	2.32
20	799140	5.6	8.2	0.017314	0.013681	0.1081	0.1727	0.0007	0.0006	673.17	1037.82	2.28	2.19
25	852416	4.5	5.6	0.009417	0.008742	0.0920	0.1258	0.0005	0.0004	523.95	752.53	1.71	1.30
30	913302	3.5	7.3	0.010823	0.014665	0.0714	0.1591	0.0002	0.0007	419.83	955.62	1.64	2.57
35	983556	4.5	8.3	0.011834	0.026774	0.0885	0.1792	0.0005	0.0008	528.86	1072.42	2.48	4.49
40	1065520	4.2	8.7	0.017740	0.016072	0.0853	0.1873	0.0005	0.0006	494.62	1120.38	2.91	2.74
45	1162385	3.9	8.5	1.051230	0.014296	0.0744	0.1850	0.0261	0.0007	450.31	1105.42	141.65	1.89
50	1278624	4.3	8.5	2.517065	0.013207	0.0782	0.1842	0.0596	0.0005	505.59	1100.53	339.00	1.78
60	1598280	3.1	8.5	4.019385	0.025161	0.0397	0.1859	0.0920	0.0006	371.29	1099.27	539.88	4.41
61	1639261	3.5	8.5	4.075541	0.050638	0.0489	0.1838	0.0933	0.0008	382.05	1095.84	546.77	5.60
62	1682400	4.3	9.2	4.013774	0.010368	0.0647	0.1966	0.0919	0.0006	461.22	1183.69	539.18	1.42
63	1727870	6.3	8.4	4.608076	2.107127	0.0509	0.1806	0.1039	0.0539	489.64	1083.30	617.03	284.82
64	1775866	4.1	8.5	4.482900	2.511354	0.0441	0.1837	0.1031	0.0632	437.36	1094.85	599.75	338.67
65	1826605	3.2	8.5	5.020343	3.739501	0.0491	0.1864	0.1142	0.0906	386.83	1104.14	669.83	502.22
70	2131040	18.8	7.9	5.530328	5.166041	0.0432	0.1742	0.1242	0.1203	696.56	1035.53	737.17	688.33

high amount of noise in the camera parameters [inputs to the BA optimization; in this case, the noise levels are indicated in Fig. 3(a)] is enough to fail the optimization. The Cauchy could optimize the camera parameters in the presence of 40% outliers, whereas the proposed robust function managed it in a much higher outlier percentage, up to 62%.

V. CONCLUSION

BA4W as a fast, robust, and efficient BA pipeline (SfM) for WAMI has been introduced. It was shown that, without neither applying a direct outliers filtering (e.g., RANSAC) nor reestimating the camera parameters (e.g., essential matrix estimation), it is possible to efficiently refine noisy camera parameters in a very short amount of time. The proposed approach is highly robust due to the proposed robust function that is adaptive with the persistency factor of each track. The proposed SfM is highly suitable for sequential aerial imagery, particularly for WAMI, where camera parameters are available from onboard sensors.

ACKNOWLEDGMENT

The authors would like to thank S. Suddarth for providing the data sets, J. Schönberger and R. Pelapur for their help in the setup and installation of Mavmap, and the anonymous reviewers for their valuable comments and suggestions. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Air Force Research Laboratory or the U.S. Government.

REFERENCES

[1] H. Stewenius, C. Engels, and D. Nistér, "Recent developments on direct relative orientation," *ISPRS J. Photogramm. Remote Sens.*, vol. 60, no. 4, pp. 284–294, 2006.

[2] J. McGlone, E. Mikhail, J. Bethel, and R. Mullen, Eds., *Manual of Photogrammetry*, 5th Ed. Falls Church, VA, USA: Amer. Soc. Photogramm. Remote Sens., 2004.

[3] B. Triggs, P. McLauchlan, R. Hartley, and A. Fitzgibbon, "Bundle adjustment—A modern synthesis," in *Proc. Vis. Algorithms: Theory Practice*, W. Triggs, A. Zisserman, and R. Szeliski, Eds., 2000, pp. 298–372.

[4] D. Nistér, "An efficient solution to the five-point relative pose problem," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 6, pp. 756–770, Jun. 2004.

[5] V. Indelman, R. Roberts, C. Beall, and F. Dellaert, "Incremental light bundle adjustment," in *Proc. British Mach. Vis. Conf.*, 2012, pp. 134.1–134.11.

[6] S. Agarwal, Y. Furukawa, and N. Snavely, "Building Rome in a day," *Commun. ACM*, vol. 54, pp. 105–112, 2011.

[7] E. Rupnik, F. Nex, I. Toschi, and F. Remondino, "Aerial multi-camera systems: Accuracy and block triangulation issues," *ISPRS J. Photogramm. Remote Sens.*, vol. 101, pp. 233–246, Mar. 2015.

[8] M. Pollefeys *et al.*, "Detailed real-time urban 3D reconstruction from video," *Int. J. Comput. Vis.*, vol. 78, no. 2/3, pp. 143–167, Oct. 2007.

[9] H. Aliakbarpour and J. Dias, "Three-dimensional reconstruction based on multiple virtual planes by using fusion-based camera network," *IET Comput. Vis.*, vol. 6, no. 4, pp. 355–369, Jul. 2012.

[10] M. Lhuillier, "Incremental fusion of structure-from-motion and GPS using constrained bundle adjustments," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 12, pp. 2489–2495, Dec. 2012.

[11] J.-M. Frahm *et al.*, "Fast robust large-scale mapping from video and Internet photo collections," *ISPRS J. Photogramm. Remote Sens.*, vol. 65, no. 6, pp. 538–549, Nov. 2010.

[12] J. Schönberger, F. Fraundorfer, and J.-M. Frahm, "Structure-from-motion for MAV image sequence analysis with photogrammetric applications," in *Proc. Int. Archives Photogramm., Remote Sens., Spatial Inf. Sci.*, 2014, vol. XL-3, pp. 305–312.

[13] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[14] S. Agarwal, N. Snavely, S. M. Seitz, and R. Szeliski, "Bundle adjustment in the large," in *Proc. Eur. Conf. Comput. Vis.*, 2013, pp. 29–42.

[15] Y. Jeong, S. Member, D. Niste, and I.-S. Kweon, "Pushing the envelope of modern methods for bundle adjustment," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 8, pp. 1605–1617, Aug. 2012.

[16] R. Lakemond, C. Fookes, and S. Sridharan, "Resection-intersection bundle adjustment revisited," *ISRN Mach. Vis.*, vol. 2013, pp. 1–8, 2013.

[17] M. Lourakis and A. Argyros, "SBA: A software package for sparse bundle adjustment," *ACM Trans. Math. Softw.*, vol. 36, no. 1, pp. 1–30, Mar. 2009.

[18] K. Konolige, "Sparse bundle adjustment," in *Proc. British Mach. Vis. Conf.*, 2010, pp. 102.1–102.11.

[19] C. Wu, S. Agarwal, B. Curless, and S. M. Seitz, "Multicore bundle adjustment," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2011, pp. 3057–3064.

[20] C. Wu, "Towards linear-time incremental structure from motion," in *Proc. Int. Conf. 3D Vis.*, Jun. 2013, pp. 127–134.

[21] M. Bryson, A. Reid, F. Ramos, and S. Sukkarieh, "Airborne vision-based mapping and classification of large farmland environments," *J. Field Robot.*, vol. 27, no. 5, pp. 632–655, May 2010.

[22] A. Albarelli, E. Rodola, and A. Torsello, "Imposing semi-local geometric constraints for accurate correspondences selection in structure from motion: A game-theoretic perspective," *Int. J. Comput. Vis.*, vol. 97, no. 1, pp. 36–53, Mar. 2012.

[23] A. Aravkin, M. Styer, Z. Moratto, A. Nefian, and M. Broxton, "Student's t robust bundle adjustment algorithm," in *Proc. Int. Conf. Image Process.*, 2012, pp. 1757–1760.

[24] S. Niko and P. Protzel, "Towards using sparse bundle adjustment for robust stereo odometry in outdoor terrain," in *Proc. TARO*, 2006, vol. 2, pp. 206–213.

[25] C. Wu, "SiftGPU: A GPU Implementation of Scale Invariant Feature Transform (SIFT)," 2007. [Online]. Available: http://cs.unc.edu/ccwu/siftgpu

[26] S. Agarwal and K. Mierle, "Ceres Solver." [Online]. Available: http://ceres-solver.org

[27] Y. Furukawa and J. Ponce, "Accurate, dense, and robust multiview stereopsis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 8, pp. 1362–76, Aug. 2010.

[28] D. Huynh, "Metrics for 3D rotations: Comparison and analysis," *J. Math. Imaging Vis.*, vol. 35, no. 2, pp. 155–164, Oct. 2009.